SciSoft - Support #23244

MicroBoone libtorch support

09/09/2019 05:33 PM - Lynn Garren

Status:	Assigned	**Start date:**	09/09/2019
Priority:	Normal	**Due date:**	
Assignee:	Lynn Garren	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour

Description

MicroBooNE is currently testing their own builds of the following packages:

- openblas v0_3_6
- libtorch v1_0_1
- pyyaml v3_12d
- opencv v3_1_0_nogui
- cppzmq v4_3_0
- libzmq v4_3_1
- pyzmq v18_0_2
- figcan v0_0_4
- SparseConvNet 8422a6f
- dllee_unified v1_0_0
- ubMaskRCNN
- ubInfillNet
- ubSSNet
- ubdl

We note that openblas, libtorch, pyyaml, and opencv are already on SciSoft.

libzmq and cppzmq are part of the zmq package which is maintained by the DAQ group. We sent an email to the DAQ group with inquiries. There is some concern that libzmq and cppzmq should be maintained as separate products.

The build of libtorch v1_0_1 will need to become v1_0_1b, since a new table file is required. Also we note that the MicroBooNE build of libtorch depends on both numpy and openblas. Since numpy is built against the lapack product, there is possible run time confusion due to duplicate function names. Taritree will try building libtorch with either lapack or eigen. This build is now really pytorch, not just the libtorch part. We are unsure if the product name should be libtorch or pytorch. It may be appropriate to use a single product name with qualifiers to indicate how it was built.

Taritree found that MicroBooNE needs an older version of opencv when using it with dllee_unified. We have requested that he create a branch in opencv. We also reqested that the version be qualified, e.g., opencv v3_1_0 -q no_gui.

Taritree and Herb will consult about ubMaskRCNN, ubInfillNet, ubSSNet, ubdl. Since these products contain data files, a ups product may not be the best solution.

dllee_unified is MicroBooNE code.

We wonder if SparseConvNet has, or will have, a version tag. The version appears to be based on a commit hash which makes simple comparisons of version numbers problematic.

This issue is a result of discussions concerning RITM0866431

History

#1 - 09/10/2019 09:13 AM - Lynn Garren

Reply from Kurt re zmq:

I believe that the only reason for bundling zmq and cppzmq together was
convenience. I'll note that the older zmq 4_1_5 version from Aaron and
Adam also had the C++ headers included with the ZMQ UPS package.
However, I don't know of any reason why they couldn't be distributed
separately. (although, early on, I did have a user request to
distribute them together, for convenience)

protoDUNE Single Phase DAQ is using ZeroMQ v4_3_1, cppzmq v4_3_0, and
pyzmq v18_0_1. So, if someone is willing to start providing
centrally-built versions of these packages, we in the DAQ will gladly
use them (e.g. we could update to pyzmq v18_0_2).

#2 - 09/16/2019 10:26 AM - Kyle Knoepfel

- Assignee set to Lynn Garren

- Status changed from New to Assigned

#3 - 09/16/2019 11:09 AM - Taritree Wongjirad

Hi.

I made a build of pytorch that uses eigen and lapack. Unfortunately, tests with existing networks show that the runtime for networks with this package increased by a large amount. For one event, the processing time went from 300 seconds using openblas to 2400 seconds using eigen/lapack. This makes eigen/lapack unusable for us or any other experiment that wishes to employ (dense) convolutional neural networks for large scale production.

As a short-term alternative, I have prepared a build of numpy and scipy that link against openblas. These products are now on uboone's cvmfs space. (Note. My openblas is compiled for sandy bridge to target the intel processors on the grid though it does run fine with the AMD pile driver CPU in the uboonebuild02 machine I use.) MicroBooNE will use this now for production tests, but we realize that this creates further issues that need to be resolved.

Bests,
Taritree